Application of Adaptive Multigrid Algorithm in HMC

Richard Brower (Boston University)

Meifeng Lin (Brookhaven National Lab)

James Osborn (Argonne National Lab)

Lattice 2014
Columbia University, June 23 – 28, 2014

BROOKHAVEN NATIONAL LABORATORY
a passion for discovery

U.S. DEPARTMENT OF ENERGY | Office of Science

Outline

- A brief review of the Hybrid Monte Carlo algorithm
- Why Multigrid in HMC and Difficulties
- Implementation Strategy
- Preliminary Results with unimproved Wilson Fermions
- Future Improvements

Hybrid Monte Carlo Algorithm

- Introduce fictitious conjugate momenta $\pi_\mu(x)$ and construct a Hamiltonian

$$H = \frac{1}{2}\pi^2 + S_{pf} + S_g,$$

where $\pi^2 = \sum_{x,\mu} \mathrm{Tr}\pi_\mu^2(x)$.

- Update the gauge links and conjugate momenta with the Hamiltonian equations of motion:

$$\begin{aligned} \dot{U}_\mu(x) &= iU_\mu(x)\pi_\mu(x), \\ \dot{\pi}_\mu(x) &= -i\left[U_\mu(x)\frac{\partial S_{eff}}{\partial U_\mu(x)}\right]_{TA} \equiv -i\left[F_\mu(x)\right]_{TA}, \end{aligned}$$

with $S_{eff} = S_{pf}[\phi,\phi^\dagger] + S_g[U]$

Force term: includes gauge and fermion parts

Pseudofermion action $S_{pf} = \phi^\dagger D^{-1}\phi$

- Accept/reject at the end of a molecular dynamics trajectory -> Metropolis step.

Matrix Inversions in HMC

- The Hasenbusch-style pseudofermion action can be generally written as

$$
\begin{aligned}
S_F[U,\{\phi_i^\dagger,\phi_i\}] \;=\;& \phi_0^\dagger \left([W_1^{-1}\hat{M}][W_1^{-1}\hat{M}]^\dagger\right)^{-1}\phi_0 \\
&+\sum_{i=1}^{n-1}\phi_i^\dagger\left([W_{i+1}^{-1}W_i][W_{i+1}^{-1}W_i]^\dagger\right)^{-1}\phi_i \\
&+\phi_n^\dagger(W_nW_n^\dagger)^{-1}\phi_n
\end{aligned}
$$

where $\hat{M}$ is the original even-odd preconditioned fermion matrix, and W_i‘s are the Hasenbusch-preconditioned matrices with heavier masses.

- For each pseudofermion field, a random vector η_i is generated. To set up the pseudofermion fields, we need to do the following:

$$
\begin{aligned}
\eta_0 &= (W_1^{-1}\hat{M})^{-1}\phi_0 \rightarrow \phi_0 = W_1^{-1}\hat{M}\eta_0, \\
\eta_i &= (W_{i+1}^{-1})^{-1}W_i\phi_i \rightarrow \phi_i = W_{i+1}^{-1}W_i\eta_i, \quad i=1,n-1, \\
\eta_n &= W_n^{-1}\phi_n \rightarrow \phi_n = W_n\eta_n.
\end{aligned}
$$

So for n Hasenbusch mass preconditioners, the setup will require n matrix inversions.

Matrix Inversions in HMC (cont'd)

- We also need to calculate the initial and final Hamiltonian before/after the Molecular Dynamics update. The fermionic part will require the following inversions:

$$\begin{aligned}
\chi_0 &= \hat{M}^{-1}(W_1\phi_0) \\
\chi_i &= W_i^{-1}(W_{i+1}\phi_i) \\
\chi_n &= W_n^{-1}\phi_n
\end{aligned}$$

- The fermionic contribution to the Hamiltonian is given by

$$S_F = \sum_{i=0}^{n} \chi_i^\dagger \chi_i$$

- The pseudofermion field setup only needs to be done at the beginning of a trajectory. The action calculation both the beginning and end of a trajectory. So the cost is negligible compared to the force calculation.

Force calculations in HMC

- The most expensive part of the HMC comes from the fermionic force term calculation.

$$F_\mu(x) = -\left[\frac{\delta S_F[U,\phi,\phi^\dagger]}{\delta U_\mu(x)}\right]^T$$

- Without Hasenbusch terms, at each step, we need to evaluate

$$\delta S_F = -\left(Q^{-1}\phi\right)^\dagger \left[(\delta\hat{M})\hat{M}^\dagger + \hat{M}(\delta\hat{M}^\dagger)\right]\left(Q^{-1}\phi\right)$$

with $Q = \hat{M}\hat{M}^\dagger$

- With Hasenbusch terms (dropping the hats on matrices), for each "ratio"

$$\begin{aligned} \delta S_{MP} &= Y^\dagger\delta W\phi - Y^\dagger\delta(MM^\dagger)Y + \phi^\dagger\delta W^\dagger Y, \\ Y &= (MM^\dagger)^{-1}(W\phi) \end{aligned}$$

- The core of the force calculations in HMC is solving

$$\boxed{MM^\dagger X = \phi}$$

Why Multigrid?

- HMC in Lattice QCD suffers from critical slowing down in several aspects:
  1. The condition number of normal operator $MM^\dagger$ becomes worse as the quark mass goes to the chiral limit, increasing the cost of inversions.
  2. The contribution to the fermion force becomes larger as the quark mass is decreased, requiring smaller step sizes and hence more integration steps and more inversions.
  3. Autocorrelation time becomes longer, requiring more trajectories in an ensemble and thus more inversions.

- The cost of HMC increases dramatically as we go to the physical limit (physical quark mass, infinite volume and continuum limit).

- Multigrid (MG) proven 10x+ more efficient than Krylov solvers such as CG and BiCGStab in the quark propagator calculations.

- It is natural to try to apply MG in HMC.

Multigrid Performance

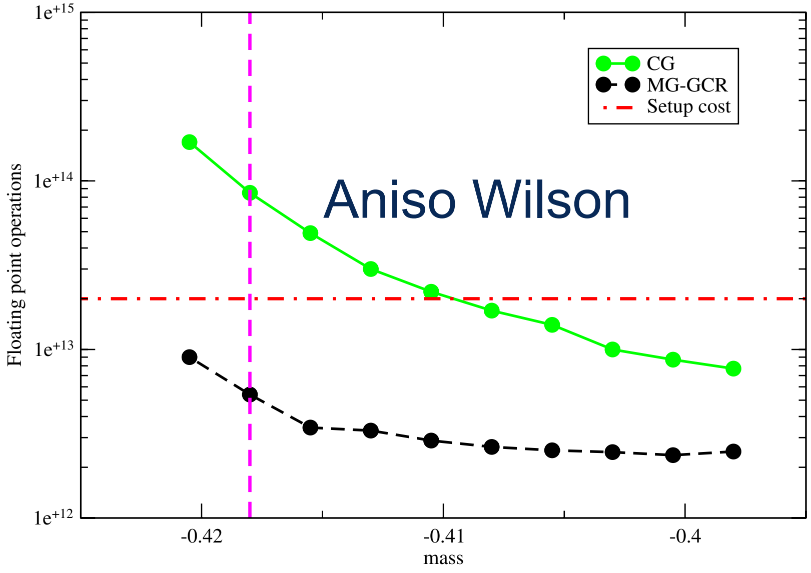


FIG. 2 (color online). Number of floating point operations required to reach convergence for CG and MG-GCR on the $V = 32^3 \times 96$ lattice (parameters given in Fig. 1). The horizontal line indicates the number of floating point operations of the MG setup.

Babich et al 2010

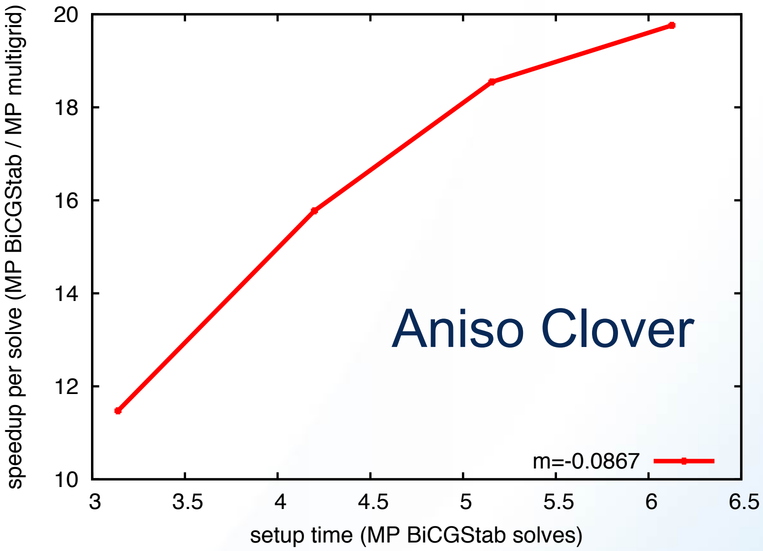


Figure 3: Speedup of multigrid solver relative to BiCGStab versus setup time at the physical quark mass.

Osborn et al 2010

- MG setup cost is several times of a BiCGStab solve.
- Must be able to reuse the setup to have performance gain. This is achieved in the propagator calculations by computing several sources per lattice.

Challenges for MG-HMC

- **Gauge fields are updated at every integration step. If we follow the same recipe as in the quark propagator calculations, we will spend way too much time doing the setup, defeating the purpose of MG.**

- Need to reduce the setup cost, or cleverly reuse the setup.

- **MG is more effective for lighter masses. With Hasenbusch mass preconditioning, the number of light quark solves has reduced dramatically. MG-HMC needs to be able to compete with the modern well-tuned HMC algorithms.**

- Need to be able to do efficient light-quark solves without much setup cost for many integration steps.
- Need to speed up the heavy-quark solves as well.
- Used on top of Hasenbusch preconditioning, it may help us gain additional factors.

Our Implementation Strategies

- Gauge field gets updated after every solve in HMC, but is highly correlated.
- The same setup may be able to be reused for several integration steps and even several trajectories if tuned well.

- Setup is done at the light dynamical mass at beginning of a trajectory.
- Reused in subsequent integration steps and/or MD trajectories until gain is lost
- Refresh the setup when **(subsequent trajectory time > setup time + 1st traj. time)**

- Use Wilson/Clover MG solver available in qopqdp.
- HMC integration done in FUEL (See James Osborn's talk at 15:35).
  - MG solver implemented for $MX = \phi$
  - Do a two-step solve for $MM^\dagger X = \phi$
    - First solve

      $$M^\dagger Y = \phi$$

    - Then solve

      $$MX = Y$$

MG Solver in qopqdp

1. Calculate the near-null vectors
   - Determine a set of N near-null vectors by applying an iterative solver to random vectors

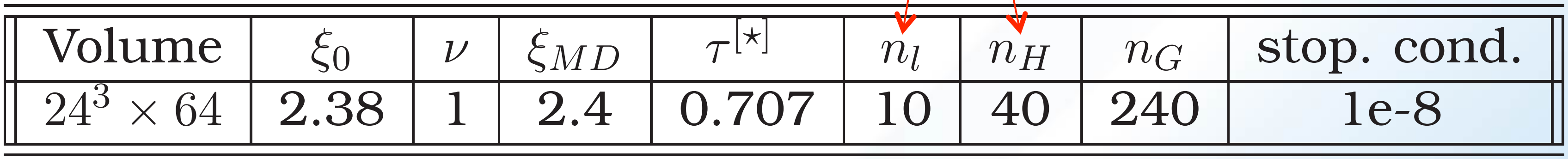
$$Mv_n = \eta_n, \quad n = 1 \dots N$$

2. Fine-level pre-smoothing
3. Restriction to construct the coarse operator.
4. Coarse-level solve.
5. Prolongation to fine level.
6. Fine-level post-smoothing.
7. Apply GCR outer solver.

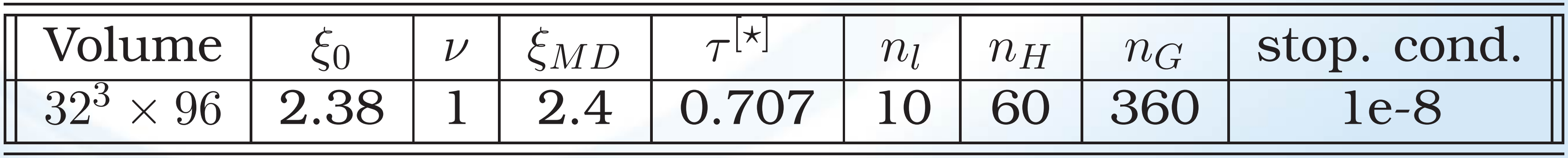
MG-HMC Tests

- Starting from existing thermalized anisotropic 2-flavor Wilson lattices. (Bulava et al. 2009)
- Apples-to-apples comparison: use the same HMC setup. Simply replace the original solver with MG solver. Has one Hasenbusch mass term.
- Pion mass ~ 420 MeV. Tested on two lattice volumes.
- Run on 32 BG/Q nodes with 32 MPI processes/node at ALCF.

m_l = -0.4125, m_H = -0.374



Volume	ξ_0	ν	ξ_{MD}	$\tau^{[\star]}$	n_l	n_H	n_G	stop. cond.
$24^3 \times 64$	2.38	1	2.4	0.707	10	40	240	1e-8



Volume	ξ_0	ν	ξ_{MD}	$\tau^{[\star]}$	n_l	n_H	n_G	stop. cond.
$32^3 \times 96$	2.38	1	2.4	0.707	10	60	360	1e-8

MG-HMC Parameter Tuning

$24^3 \times 64$

MG parameters	Run 1	Run 2	Run 3	Run 4	Run 5 [nvecs=16]
setup_res.	0.4	0.1	0.4	0.1	0.5
cres	0.3	0.3	0.5	0.3	0.3
setup_change_fac	0.4	0.1	0.2	0.4	0.4
npre	5	5	4	0	5
npost	9	9	9	5	9
scale	1	0	0.2	1	1
Setup Time [secs]	49	61	43	65	29
Traj. 1 Time [secs]	160	575	308	152	162
Traj. 2 Time [secs]	182	672	376	214	176
Traj. 3 Time [secs]	201	686	409	282	192
Traj. 4 Time [secs]	222	681	427	344	208

- Many parameters to tune.
- Fixed nvecs = 24 in Run 1-4, and 16 in Run 5.
- Scanned other parameters to find the best set.

Tuning for MG-HMC

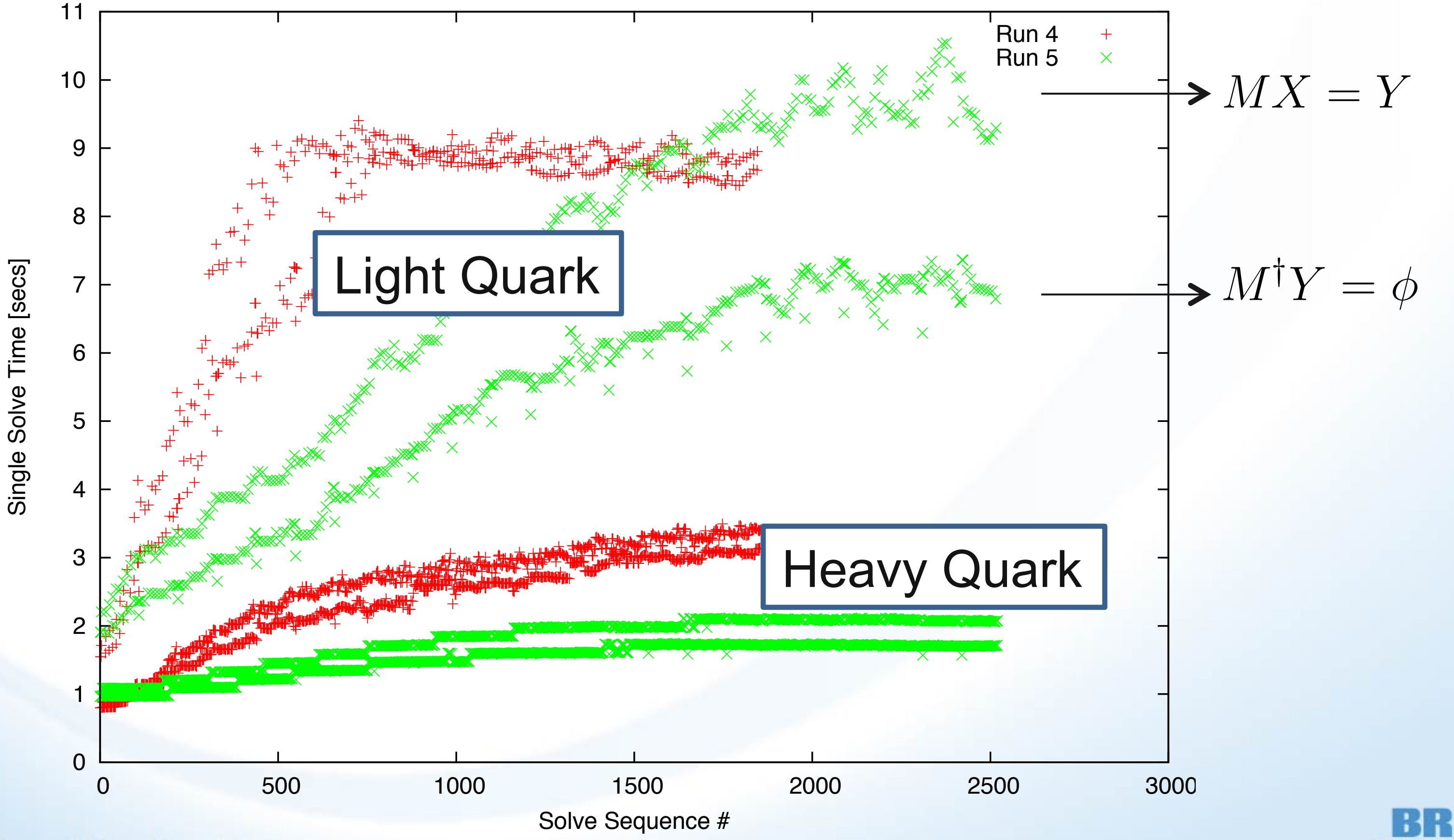
$24^3 \times 64$

MG parameters	Run 1	Run 2	Run 3	Run 4	Run 5 [nvecs=16]
setup_res.	0.4	0.1	0.4	0.1	0.5
cres	0.3	0.3	0.5	0.3	0.3
setup_change_fac	0.4	0.1	0.2	0.4	0.4
npre	5	5	4	0	5
npost	9	9	9	5	9
scale	1	0	0.2	1	1
Setup Time [secs]	49	61	43	65	29
Traj. 1 Time [secs]	160	575	308	152	162
Traj. 2 Time [secs]	182	672	376	214	176
Traj. 3 Time [secs]	201	686	409	282	192
Traj. 4 Time [secs]	222	681	427	344	208

- Run 4 has the best time for first trajectory, but deteriorates quickly
- Run 5 has the best overall performance.
- Same setup can be used for 3 trajectories.

An Optimization Problem

- If the setup is tuned too well for the first solve, subsequent solves get worse quickly.
- If it is not tuned well, overall gain is small.
- It is tricky to find the sweet spot.



$$\rightarrow MX = Y$$

$$\rightarrow M^{\dagger}Y = \phi$$

MG-HMC Performance

24^3x64, time averaged over 20 trajectories

Solver	Light Solve [secs]	Heavy Solve [secs]	Trajectory Time [secs]
CG	176	121	326
BiCGStab	111	99	239
MG	61	91	187

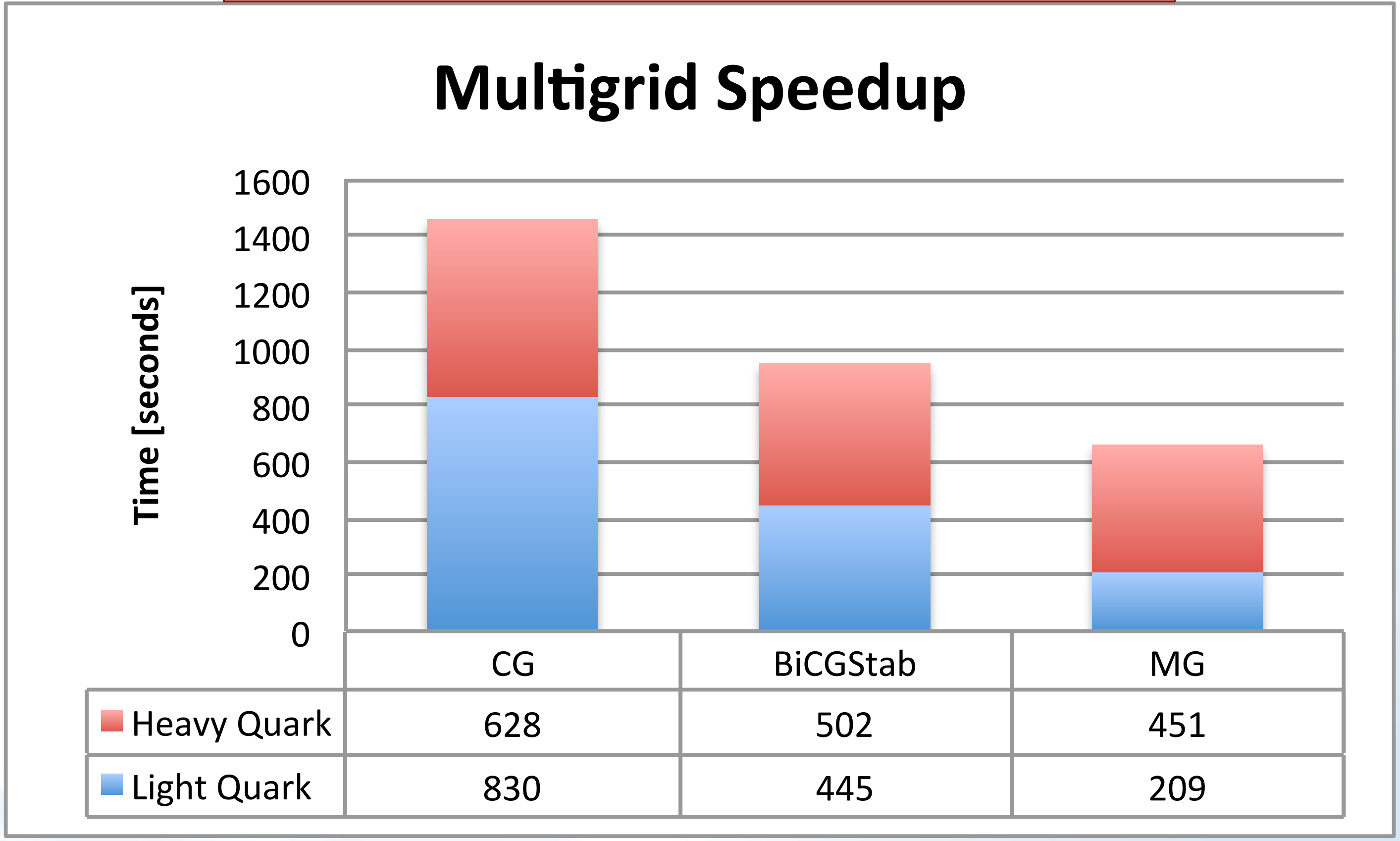
32^3x96, time averaged over 3-8 trajectories

Solver	Light Solve [secs]	Heavy Solve [secs]	Trajectory Time [secs]
CG	830	628	1596
BiCGStab	445	502	1086
MG	209	451	822

- Light solve: MG is 2x faster than BiCGStab, 3-4x faster than CG
- Speedup per trajectory not as big
- Bottleneck is heavy solves ➔ Can rebalance HMC

MG-HMC Speedup

32^3 x 96
420 MeV pion
Wilson fermions



	CG	BiCGStab	MG
🟥 Heavy Quark	628	502	451
🟦 Light Quark	830	445	209

Reversibility

- Will reusing the setup affect reversibility?
- No sign so far, but more tests are needed.

```
==With MG==
Sold: 22721701.88   Srev: 22721701.88   dS: 1.329928637e-06
Sold: 22725067.11   Srev: 22725067.11   dS: -0.001061491668
Sold: 22713290.68   Srev: 22713290.68   dS: 0.0005583688617
Sold: 22721697.35   Srev: 22721697.35   dS: -0.0001310259104
Sold: 22724432.14   Srev: 22724432.14   dS: -0.0001665465534

==Without MG (BiCGStab)==
Sold: 22721701.88   Srev: 22721701.88   dS: 0.0003642588854
Sold: 22725067.1    Srev: 22725067.1    dS: -0.0002857670188
Sold: 22713290.69   Srev: 22713290.69   dS: -0.0004257671535
Sold: 22721697.35   Srev: 22721697.35   dS: -0.0006039328873
Sold: 22724432.15   Srev: 22724432.15   dS: -0.0003919377923
```

Future Improvements

- Implement it for clover fermions. ➔ Needs the clover force term.

- Test with lighter masses and larger lattices. Ultimately that's where MG will help the most.

- Reduce setup cost:
  - Subsequent setups may be able to start from the previous null vectors to save time in the calculation of null vectors.

- Investigate more MG levels.

Conclusions

- With proper tuning, MG can further speed up the modern well-tuned HMC algorithms.

- Initial tests on anisotropic 2-flavor Wilson lattices with a pion mass of 420 MeV show that the speedup with MG-HMC compared to Hasenbusch-preconditioned HMC can be 2X.

- More tests with lighter masses on larger lattices are needed.

- ➢ MG parameters auto-tuning?